**Decisiones de diseño – Notesbox**

Un cliente puede crearse una cuenta, detallando sus diferentes datos incluyendo la direccion. Esa cuenta ademas posee un carrito donde puede ir agregando diferentes cajas.

Cada caja tiene una melodia asociada, y cada melodia tiene una categoria especifica.

La categoria es un ENUM para mantener una coherencia de nombres (evitar errores de tipeo).

La melodia tambien tiene una secuenciaMIDI que esta modelada como un String (no estoy seguro como representar un archivo MIDI).

Un carritoCompras contiene una lista de cajas.

Un cliente tambien tiene una lista de ordenes que puede generar con su carrito (incluye a todas las cajas del carrito).

Una orden tiene además un estadoOrden y la fabrica a la que pertenece.

Un cliente puede agregar una caja al carrito, generar una orden y mostrar/listar sus ordenes.

Por ahora modelé a la API que usa nuestro sistema para el stock con una interfaz “ServicioStock”.

El cliente haría uso de ServicioStock cuando genera una orden.

Respecto al Script para convertir un MIDI a una Especificacion, metí esa lógica (desconocida) dentro de una clase “ConvertorMIDIaEspecificacion”, que tiene un metodo que usa ese Script mencionado en el enunciado.

La especificacion la modele como una clase, pero tranquilamente podría ser un String. Solo lo hice para que quede claro que estoy devolviendo un dato particular, en este caso una Especificacion.

El Admin tiene una serie de métodos que reflejan todas las funcionalidades que dice el enunciado, haciendo uso o devolviendo Ordenes, el ConvertorMIDI y la Especificacion. No me puse a ver a detalle la lógica interna de cada método.